Sampled Data Digital Filtering System

PER NOA DATED 6-22-04

This is a non-provisional application of provisional application serial No. 60/121,198 by D. McNeely, filed February 24, 1999.

Field of the Invention

This invention is related to the sample rate conversion of digital data for data processing applications including video and audio among others.

Background of the Invention

There are a variety of applications requiring conversion of digital data samples occurring at a first data rate to digital data samples occurring at a different second data rate. In these applications, data sampled at one rate is interpolated to provide estimates of data at a different rate or sampling phase. Sample rate conversion applications include, for example, conversion between different video standards such as between High Definition TV (HDTV) data and CCIR601 standard data and conversion between different video display formats such as between interlace and progressive display formats. Other applications include, for example, multi-media composite image formation and display e.g. for Picture-in-Picture (PIP) presentation, and data processing for digital data storage such as for CDROM or DVD applications and digital demodulation involving digital sample rate conversion for establishing frequency, phase or symbol timing synchronization. The widespread use of sample rate conversion in cost sensitive applications means that there is a need to optimize configurations for both sample rate converters and their interpolator and digital filter sub-components.

Known, classical interpolator configurations have been thought to be optimal in terms of maximizing performance and minimizing the hardware complexity involved, i.e., the numbers of adders, multipliers, registers etc. that are used. However, the sample rate converter, interpolator and digital filter systems and the methods of their derivation presented herein provide improvements over the known filter designs in both performance and reduced complexity.

Summary of the Invention

A digital filtering system for filtering sample data includes a delay network for delaying input sample data to provide multiple delayed sample data

outputs. The filtering system also includes a filter network representable by a decomposed (structurally factored) coefficient weighting matrix for processing the delayed sample data outputs. A processor produces a filtered output by computing a weighted product summation of the delayed sample data outputs and the coefficient weighting matrix.

Brief Description of the Drawings

In the drawing:

Figure 1 shows an enhanced performance sample rate converter for converting the horizontal line sample rate of luminance and chrominance data samples in a video display format conversion application, according to the principles of the invention.

Figure 1A shows an enhanced performance digital filter for use as digital filter 40 in the sample rate converter configuration of Figure 1, according to the principles of the invention.

Figure 2 shows a filter configuration suitable for providing the $H_1(z)$ and $H_0(z)$ outputs of the first interpolation network stage bootstrap filter of unit 10 of Figure 1, according to the invention.

Figure 3 shows a reduced complexity digital filter for use as digital filter 40 in the sample rate converter configuration of Figure 1, according to the principles of the invention.

Figure 4 shows a table indicating coefficient values and the effective filter delay in conjunction with the corresponding position index signal for the reduced complexity digital filter of unit 40 of Figure 3.

Figure 5 shows an arrangement exemplifying extension of the interpolation network (unit 10) and delay network (unit 20) of Figure 1 to provide enhanced interpolator performance, according to the principles of the invention.

Figure 6 shows the principal elements of a structural factoring process used to derive improved digital filter functions, according to the principles of the invention.

Detailed Description of the Drawings

Figure 1 shows an enhanced performance sample rate converter for converting the horizontal line sample rate of luminance and chrominance data samples in a video display format conversion application. Although the disclosed converter is described in the context of a system for processing video signals for display format conversion purposes e.g. to upsample a standard definition format of 720x1280 pixel resolution to a high definition format of 1080x1920 pixel resolution (or vice versa), it is exemplary only. The converter and digital filter configurations and inventive principles disclosed may be used for any filtering or sample rate converter application including either upsampling or downsampling conversions.

In overview, the sample rate converter system of Figure 1 comprises compensation pre-filter 17, interpolation network 10, delay network 20 and digital filter 40. Pre-filter 17 is optional and used to optimize performance in terms of noise rejection outside of the desired passband and gain within the desired passband in the high performance system of Figure 1. In contrast, in a reduced complexity sample rate converter system (described later in connection with Figure 3), pre-filter 17 is omitted.

A sample rate converter converts data at an input sample rate to data at a different output sample rate and possesses a number of mutually dependent properties. These include, for example, gain and phase response characteristics, phase delay, group delay, and clock delay. In order to achieve a desired sample rate converter performance and to tailor converter operation for a particular application, it is necessary to select an acceptable compromise between these properties, both for the converter as a whole and for intermediate processing stages within the converter. In deriving a sample rate converter configuration, individual selected properties may be established as either invariant properties or as properties to be optimized.

Important objectives in sample rate converter design are: 1) maximizing performance including minimizing alias (i.e. interference) components in the Pass Band, and 2) minimizing complexity (e.g. measured in the number of adders required for an implementation). Normally performance and complexity are inversely related in sample rate converter design. However, by viewing a sample rate converter system as being equivalent to a multi-stage interpolator comprising a sequence of cascaded interpolators, an advantageous sample rate converter configuration is derived offering reduced complexity and improved performance over previously deemed optimal arrangements. In such a multi-stage interpolator, a first "coarse" interpolator stage converts a spatial sampling grid of an input signal to a higher resolution spatial sampling grid (a "coarse" up-sample grid). A second "finer" resolution interpolator stage converts the coarse up-sample grid to a spatial sampling grid of desired

resolution. A conventional implementation of this concept would require three clock domains representing the input sample rate, an intermediate re-sampled rate and the desired output re-sampled rate. The sample rate converter system disclosed herein and exemplified in Figure 1 provides data at the desired output sample rate whilst advantageously:

1. processing data at a reduced number of data rates, specifically at two data rates (the input sample rate and the desired output sample rate), and
2. maximizing the proportion of circuitry operating at the input sample rate (input sample rate is less than the output sample rate for an upsample application and greater than the output sample rate for a down-sample application).

These important benefits are achieved by employing advantageous sample rate converter architectures that merge and integrate the coarse and fine interpolator stages of the multi-stage interpolator. Such an advantageous architecture is exemplified by the compensation pre-filter 17, interpolation network 10, delay network 20 and digital filter 40 of Figure 1. In such an improved converter architecture a second "finer" resolution interpolator stage is chosen to have a reduced number of intermediate delay stages (taps) with a larger number of sets of digital filter weighting coefficients than is typical for an equivalent conventional converter architecture. Note, each digital filter weighting coefficient is associated with an intermediate delay stage. The "coarse" first stage interpolator is chosen to have a larger number of intermediate delay stages (taps) with a lower number of sets of digital filter weighting coefficients than is typical for an equivalent conventional converter architecture.

An improved converter architecture using an integrated multi-stage interpolator according to invention principles includes:

1. A second stage interpolator (exemplified by digital filter 40 of Figure 1) of finer resolution than a first stage interpolator (units 10 and 20 of Figure 1) that is configured to isolate its tapped delay line which comprises multiple intermediate delay stages.

2. A generalized tapped delay line (exemplified by interpolation network 10 in conjunction with delay network 20 of Figure 1) replacing the isolated tapped delay line of the second stage interpolator. This generalized tapped delay line is clocked at the input sample rate and consists of intermediate delay stages (taps) with a larger number of sets of digital filter weighting coefficients than is typical for an

equivalent conventional converter architecture. The generalized tapped delay line includes:

a) a first interpolation network (unit 10 of Figure 1 comprising a temporal bootstrap filter), for interpolating data at a first sample rate to provide upsampled interpolated data samples according to a first sample spacing and having one input and n outputs,

b) multiple tapped delay lines (units 22, 24 and 26 of delay network 20 of Figure 1) comprising a delay network providing delayed upsampled interpolated data samples according to a second sample spacing of higher resolution than the first sample spacing. Sets of delayed upsampled interpolated data samples encompass (i.e. precede and succeed) a corresponding original sample position, and

c) a multiplexer network (units 27, 29, 31 and 33 of delay network 20 of Figure 1) providing multiple $\frac{T}{n}$ spaced delay line outputs (taps) comprising higher resolution second sample spacing data surrounding the desired output sample time, where T is the period between samples of the input sampled rate and n, in the converter architecture of Figure 1, is 2.

In considering the operation of the sample rate converter of Figure 1 in detail, a pre-filtered compensated input signal is provided by unit 17 to units 13 and 15 of interpolation network 10. The signal provided by unit 17 is filtered with a transfer function selected to optimize overall passband performance of the Figure 1 converter in terms of providing noise rejection outside of the passband and to provide an optimized (i.e. ideally flat) gain response within the passband. Units 13 and 15 of network 10 process the data from unit 17 and provide upsampled data to delay network 20. Specifically, units 13 and 15 interpolate the input data from unit 17 with transfer functions $H_0(z)$ and $H_1(z)$ respectively and provide interpolated data outputs to delay network 20. Unit 13 ($H_0(z)$) in the embodiment of Figure 1, merely passes the original pre-filtered data from unit 17 to delay network 20 i.e. $H_0(z)$ in the embodiment of Figure 1 is a delay function. Unit 15 ($H_1(z)$) in the embodiment of Figure 1 interpolates the sampled data from unit 17 to provide intervening data samples intermediate between the input samples from unit 17. Therefore, the two outputs provided by units 13 and 15 together comprise data representative of the pre-filtered input data upsampled by a factor of two. In other embodiments units 13 and 15

provide interpolated data, that is upsampled or downsampled by the desired sampling factor, to delay network 20 which may employ transfer functions that are either the same, or different, as desired.

The data from unit 13 of interpolation network 10 is provided to multiplexer 33 via both delays 24 and 26 and to multiplexer 27 via delay 24 and also to multiplexers 29 and 31. The data from unit 15 of interpolation network 10 is provided to multiplexers 31 and 33 via delay 22 and to multiplexers 27 and 29. In this configuration, units 22, 24 and 26 of network 20 comprise multiple tapped delay lines providing outputs of delayed upsampled interpolated data samples of higher resolution than the sample spacing of the input data from unit 17. Multiplexers 27, 29, 31 and 33 multiplex the inputs received from units 10, 22, 24 and 26 to provide a selected set (selected from between two available sets) of upsampled delayed samples to digital filter 40.

Multiplexers 27, 29, 31 and 33 multiplex between the two sets of upsampled delayed inputs from units 10, 22, 24 and 26 in response to a position representative selection signal identifying the upsampled delayed output sample set spatially encompassing (i.e. straddling) the corresponding position of the converter output sample whose value is currently being determined by the converter of Figure 1. Specifically, this selection signal identifies and selects the set of four upsampled delayed output samples comprising the two upsampled delayed output samples located either side of the corresponding output sample being determined by the Figure 1 converter system. In the configuration of Figure 1, the selection signal input to multiplexers 27, 29, 31 and 33 comprises the MSB (most significant bit) of a position index signal used by filter 40 to spatially interpolate between two input samples being processed.

The set of four upsampled delayed output samples provided by network 20 to digital filter 40 consist of multiple $\frac{T}{n}$ spaced delay line outputs (taps) comprising higher resolution second sample spacing data surrounding the output sample time desired (where T is the period between samples of the input sample data from unit 17 and n is 2 in the architecture of Figure 1).

Other architectures with other values of n may be derived by replacing an isolated tapped delay line with the advantageous generalized delay line arrangement in accordance with the invention principles. For example, in the arrangement of Figure 5 (discussed later), the generalized delay line of Figure 1 is extrapolated to provide n=3. Further, the use of the generalized delay line configuration of Figure 1 facilitates the processing of the input sample data at a single data rate. Specifically, the configuration of Figure 1 enables filter 17, interpolation network 10 and delay network

20 to process input data at the input sample rate. The use of the generalized delay line maximizes the proportion of converter system circuitry operating at the input sample rate and allows a concomitant reduction in the quantity of converter circuitry that is required to implement the sample rate conversion function (in comparison to a conventional approach). Further, the use of the generalized delay line arrangement advantageously enables the processing of data in the converter of Figure 1 at two data rates, specifically at the input sample rate and at the desired output sample rate.

The sets of four upsampled delayed output samples provided by network 20 to the second stage fine resolution interpolator of digital filter 40 are interpolated by filter 40 to provide sample data at the desired output sample rate. In providing the output sample data at the desired output sample rate, unit 40 processes the upsampled delayed output sample sets from unit 20 together with a spatial position index signal used by filter 40 to spatially interpolate between two input samples being processed.

Figure 1A shows the structure of an exemplary enhanced performance digital filter for use as digital filter 40 in the sample rate converter configuration of Figure 1. Filter 40 (Figure 1A) advantageously implements a fine resolution second stage interpolation function with reduced circuitry (i.e. a reduced number of adders). The interpolation function performed by filter 40 is derived using an advantageous method of mathematically structuring and factoring the desired filter function into a minimized logic implementation. This advantageous filter function derivation method is arrived at from the consideration that a sample rate converter system may be viewed as comprising a pre-filter, a sample rate converter and a post-filter in a cascade that possesses inter-related characteristics. These characteristics include, for example, gain and phase response characteristics, phase delay, group delay, and clock delay for the converter as a whole and for intermediate processing stages within the converter.

The method of mathematically structuring and factoring the desired filter function into a mathematical function yielding a minimized logic implementation derives, in part, from the realization that advantage may be taken of relaxing certain filter characteristics to achieve simplification in implementing the filter function. However, such mathematical structuring is constrained by the desired performance requirements of the filter and the consequent need to maintain particular characteristics invariant during the factoring process. In the video processing application for which the Figure 1A converter is intended, filter 40 is desired to minimize alias components in the Passband (ideally to less than 50 dB) whilst also maintaining low complexity (measured in number of adders required for the filter 40 implementation). Normally performance and complexity are inverse relationships in filter design (an interpolator can be viewed as a programmable filter). However, both objectives may be met by

structurally factoring a mathematical interpolation function as disclosed herein to yield an optimally minimized circuit implementation of a desired filter function that is tightly tied to the mathematical structure of the desired filter function.

Figure 6 shows the principal elements of a structural factoring process used to derive improved digital filter functions. The structural factoring process comprises structuring a mathematical expression representing a filter function into a form that maps into a linear sequence of operations that are readily implementable in logic (i.e. adders, multipliers and latches). The factoring process of Figure 6 comprises determining delay (e.g. achieved using a shift register), linear transformation and factor combination processes (steps 605, 610 and 615 respectively). The structural factoring steps 605, 610 and 615 are described as follows.

1. The shift register process 605 comprises representing an interpolator's output in column vector form,

$$S^k = \{z^{-k}, k=0..M\} = \text{transpose}([1, z^{-1}, z^{-2},\ldots,z^{-M}]),$$

where the interpolator's output at any one time is a function of M+1 contiguous input samples.

2. The linear transform process 610 comprises representing series and parallel connections of a multi-input multi-output filter network as both a single matrix, L_k^i (where i is row index (superscript), k is column index (subscript)), and products and sums of matrices. It is through decomposing the single matrix L_k^i into a matrix expression that an optimally minimized filter hardware implementation (in terms of latches, adders and multipliers) is achieved.

3. The factor combiner process of step 615 derives the filter 40 (Figure 1A) output by calculating a weighted sum of the N outputs ("factors") of the linear transform process of step 610. For the converter of Figure 1A the factor combiner is constrained to employ a single parameter u representing interpolate spatial position by the row vector,

$$U_\iota = \{u^{-i}, i=0..M\} = [1, u, u^2, \ldots, u^{(N-1)}]$$

Given the conditions that, u=0 corresponds to a delay of half way between two input samples, u is within (-0.5,0.5), and M is odd, then matrix L_k^i has the following properties; odd rows are coefficient symmetric and even rows are coefficient anti-symmetric. An L_k^i matrix for a four tap filter (M=3) comprising filter

40 of Figure 1A that provides a minimum alias solution for 0 to 0.6 of the Nyquist folding frequency, is closely approximated with the integer matrix:

$$L_k^i = \frac{\begin{bmatrix} 6 & 58 & 58 & 6 \\ 23 & 59 & -59 & -23 \\ 31 & -31 & -31 & 31 \\ 16 & -48 & 48 & -16 \end{bmatrix}}{128}.$$

The number of adders ($\# L_k^i$) required to implement each constant are

$$\# L_k^i = \begin{bmatrix} 1 & 2 & 2 & 1 \\ 2 & 2 & 2 & 2 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}, \text{ Plus 12 adders for accumulation}$$

A direct implementation of this filter would take 32 adders but taking advantage of individual row symmetries could reduce this to 21 adders. However, by applying the structural factoring method and decomposing the single matrix L_k^i into a matrix expression, further minimization in filter hardware is achieved. Such structural factoring takes advantage of the existence of a factor comprising a sparse matrix in matrix L_k^i that results from matrix row symmetries. A sparse matrix is a matrix containing multiple zero value elements. While the constants that fill this sparse matrix are dependent upon a particular matrix L, it is the row coefficient symmetry imposed upon L_k^i that ensures that a sparse matrix factor with at least half of its entries of zero value exists. Many solutions with this form exist for a specific L_k^i. One of minimal implementation complexity is constructed as follows. The filter 40 matrix L_k^i is structurally factored to provide:

$$L_k^i = \frac{\begin{bmatrix} 64 & 0 & 6 & 0 \\ 0 & 128 & 0 & 23 \\ 0 & 0 & 31 & 0 \\ 0 & 0 & 0 & 16 \end{bmatrix}}{128} \cdot \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 1 & -1 & 0 \\ 1 & -1 & -1 & 1 \\ 1 & -3 & 3 & -1 \end{bmatrix}$$

$$\# L_k^i = (0+0+2+2) + 2_{acc} + (0+1+1+0) + 8_{acc} = 16 \text{ adder}$$

(Where $\# L_k^i$ is the number adders required for implementation).

Therefore, structural factoring yields a 50% reduction in the number of adders required for this factored implementation in comparison to the direct implementation

(i.e. 16 adders are required vs. 32 adders). Filter 40 of Figure 1A implements this structurally factored solution which represents the function:

$H(DC)=1,$

$$H_{4tap}(z)=\begin{bmatrix}1 & \mu & \mu^2 & \mu^3\end{bmatrix}\cdot\begin{bmatrix}\frac{1}{2} & 0 & \frac{3}{64} & 0\\ 0 & 1 & 0 & \frac{23}{128}\\ 0 & 0 & \frac{31}{128} & 0\\ 0 & 0 & 0 & \frac{1}{8}\end{bmatrix}\cdot\begin{bmatrix}0 & 1 & 1 & 0\\ 0 & 1 & -1 & 0\\ 1 & -1 & -1 & 1\\ 1 & -3 & 3 & -1\end{bmatrix}\cdot\begin{bmatrix}1\\ z^{-1}\\ z^{-2}\\ z^{-3}\end{bmatrix}$$

ASSEMBLY FUNCTIONS BASIS FIRs

$$\mu\in[-.5,.5]$$

Filter 40 advantageously implements this minimized structurally factored function to provide the Figure 1A sample rate converter output whilst processing data at a single data rate (the desired output sampling rate). It is implemented with three multipliers (units 43, 46 and 49) and 16 adders (units 51-81) and 39 latches as depicted in Figure 1A. This filter function provides improved performance in terms of reducing passband alias components (critical for video processing type applications) in comparison with a conventional function. The improved performance is achieved with reduced circuit hardware cost and complexity. The structural factoring method is also applicable for optimizing and minimizing of other digital filter functions for sample rate converter and other uses.

The structurally factored filter function of unit 40 (and other filter functions) may be further simplified by taking advantage of other row properties. This may be done in the function of filter 40, for example, by making use of an additional matrix factor and by decomposing matrix L_k^i into the following.

$$L_k^i=\frac{\begin{bmatrix}64 & 0 & 6 & 0\\ 0 & 128 & 0 & 23\\ 0 & 0 & 31 & 0\\ 0 & 0 & 0 & 16\end{bmatrix}}{128}\cdot\left[\begin{bmatrix}0 & 1 & 1 & 0\end{bmatrix}\otimes\begin{bmatrix}0 & 1 & 0\\ 1 & 0 & -1\\ 1 & -2 & 1\end{bmatrix}\right]$$

$\#L_k^i=(0+0+2+2)+2_{acc}+(0+0+0+0)+5_{acc}=$ 11 Adders

(Where $\#L_k^i$ is the number adders required for implementation).

Therefore, further structural factoring yields a 66% reduction in the number of adders required for this factored implementation in comparison to the direct implementation (i.e. 11 adders are required vs. 32 adders).

Figure 2 shows a filter configuration suitable for providing the $H_1(z)$ and $H_0(z)$ outputs of the first interpolation network stage (bootstrap filter) of units 13 and 15 within unit 10 of Figure 1. As previously mentioned, function $H_0(z)$ is merely a pass-through identity function ($H_0(z)=z$) whereby input pre-filtered data (from unit 17 of Figure 1) is buffered in unit 200 of Figure 2 before being provided to delay network 20 (Figure 1). The Figure 2 interpolator function $H_1(z)$ interpolates the input sampled data after buffering by unit 200 to provide intervening data samples intermediate between the input samples from unit 17. The $H_1(z)$ interpolator function, comprising adders (units 204, 206 and 214-232), scalers (units 202, 208, 210, 212, 234 and 236) and output stage 250. The $H_1(z)$ function is shown as follows.

$$H_1(z)=\frac{2-5z^{-1}+11z^{-2}-24z^{-3}+80z^{-4}+80z^{-5}-24z^{-6}+11z^{-7}-5z^{-8}+2z^{-9}}{128}$$

The resultant outputs of the $H_0(z)$ and $H_1(z)$ functions, provided by the Figure 2 arrangement, comprise interpolated data samples upsampled by a factor of two in comparison with the input data.

Figure 3 is an example of an advantageous reduced complexity filter (in comparison to the high performance filter of unit 40 of Figure 1A) that may alternatively be used in the sample rate converter configuration of unit 40 of Figure 1. The filter of Figure 3 employs nine adders (units 312, 314, 318, 334, 340, 349, 374, 380 and 392) and two multipliers (units 326 and 352) plus a plurality of D-type register delay stages and scaling and other stages. (Note, units 320, 342 and 386 are not counted as adders for circuit purposes as they merely add a digital one value and may be implemented without additional adder elements). The filter of Figure 3 is used for interpolation and may be used to perform a wide range of sample rate conversions. The filter of Figure 3 does this by calculating the closest interpolate value to each value at a required output spatial position using a x32 spatial upsampling output grid. The filter of Figure 3 implements the following interpolation function.

$$H_{4tap}(z)=\begin{bmatrix}1 & \mu & \mu^2\end{bmatrix}\cdot\frac{\begin{bmatrix}0 & 0 & 3 & 0\\ -1 & 4 & -2 & -1\\ 1 & -1 & -1 & 1\end{bmatrix}}{3}\cdot\begin{bmatrix}1\\ z^{-1}\\ z^{-2}\\ z^{-3}\end{bmatrix},\ H(DC)=1$$

The structural factoring method may be applied to this function to decompose the single matrix L_k^i into a matrix product expression and obtain further minimization in filter hardware. Such structural factoring takes advantage of the existence of a factor comprising a sparse matrix in matrix L_k^i that results from matrix row symmetries. Applying the structural factoring method to take advantage of row symmetries and sparse matrix factor, the following function is derived.

$$H_{4tap}(z)=\begin{bmatrix}1 & \mu & \mu^2\end{bmatrix}\cdot\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & -1\\ 0 & 0 & 1\end{bmatrix}\cdot\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & \frac{1}{3}\end{bmatrix}\cdot\left[\begin{bmatrix}0 & 0 & 1 & 0\end{bmatrix}\otimes\begin{bmatrix}1 & -1\end{bmatrix}\right]\cdot\begin{bmatrix}0 & 1 & 0\\ 1 & 0 & -1\end{bmatrix}\cdot\begin{bmatrix}1\\ z^{-1}\\ z^{-2}\\ z^{-3}\end{bmatrix},\ H(DC)=1$$

1 adder + 2 adder + 2 adder = 5 adder Linear Transform

This advantageous configuration of the Figure 3 filter function may be implemented with three less adders than the non-structurally factored version. This represents a significant saving in the required circuit hardware.

Figure 4 shows a table indicating coefficient values and effective filter delay in conjunction with the corresponding position index signal for the reduced complexity digital filter of unit 40 of Figure 3. Figure 4 lists the coefficient values, effective filter delay values and position index values for each of the 32 spatial sample positions used by the Figure 3 interpolation filter in its x32 spatial upsampling output grid. The position index data 420 (in the first column of Figure 4) is used by filter 40 to spatially interpolate between two input samples being processed. The position index signal 420 controls the phase of filter 40 on a pixel by pixel basis. It does this through multipliers 326 and 352 (via delays 302-308 of Figure 3) and by selecting between two sets of delayed upsampled interpolated data samples encompassing a corresponding original sample position via the multiplexers of unit 20 (Figure 3).

The delay data 425 in the second column of Figure 4 shows the actual effective delay through filter 40. The delay values range from 48/32 (Figure 4 column 425 item 0) to 17/32 (Figure 4 column 425 item 31). Individual delay values are expressed as a fraction of the input sample period and consequently range from 1.5 to

approximately 0.5 input samples in duration. The delay data 425 shows the actual effective delay through filter 40 corresponding to each of the two sets of delayed upsampled interpolated data samples being processed. Selection between the two sets of delayed upsampled interpolated data samples being processed and the corresponding filter 40 delay is determined by position index signal 420. The processing of the first set of delayed upsampled interpolated data samples from the multiplexers of unit 20 (Figure 3) involves data items 425-445 presented in the first 16 shaded spatial interpolation positions (position index 420-items 0-15) of Figure 4. The processing of the second set of delayed upsampled interpolated data samples from the unit 20 multiplexers (Figure 3) involves data items 425-445 presented in the second 16 non-shaded spatial interpolation positions (position index 420-items 16-31) of Figure 4.

The next four columns 430-445 of Figure 4 show the effective weighting coefficients of the four taps of the converter of Figure 3. The first two of these columns 430 and 435 (1 for no delay, and $z^{-0.5}$ for a one half clock delay) actually show the values of the gain produced by coefficients C0 and C1 of the Figure 3 interpolation filter. The last two columns 440 and 445 (z^{-1} - a one clock delay, and $z^{-1.5}$ - a one and one half clock delay) show the gains for the last coefficient C2 and C3 stages. Note that these are effective coefficients and cannot be individually localized in the hardware.

Figure 5 shows an arrangement exemplifying extension of the interpolation network (unit 10) and delay network (unit 20) of Figure 1 to provide enhanced interpolator performance. Specifically, Figure 5 shows how the generalized delay network of units 10 and 20 of Figure 1 may be replaced by corresponding interpolation network 510 and delay network 520 to provide higher resolution interpolation. Figure 5 also shows how virtually any tapped delay line (e.g. delay line 505) used in any digital filter or sample rate converter system, for example, may be replaced by the advantageous generalized delay line exemplified by units 10 and 20 of Figure 1 and units 510 and 520 of Figure 5. Such replacement yields improvements including (a) increasing the proportion of sample rate converter circuitry operating at the input sample rate, and (b) enabling a consequent reduction in the number of data rates required by sample rate converter system. In the system of Figure 1, for example, data may be processed at two data rates (the input sample rate and the desired output sample rate). This eliminates the need for processing data at an intermediate data rate as is typically required by a system using the conventional delay line 505 (Figure 5) configuration.

In the generalized delay line of Figure 5, interpolation network 510 comprises digital FIR filters H0, H1, H2 having respective impulse responses that

provide three distinct output signals each with a sample spacing equal to the input sample spacing. However, the three output signals from filters H0, H1 and H2 are phase shifted with respect to each other and together represent an upsampling of the input signal by a factor of three. The three output signals from units H0, H1 and H2 of unit 510 are multiplexed by delay network 520 to providing multiple $\frac{T}{n}$ spaced delay line outputs (taps) comprising higher resolution sampled data surrounding the output sample time desired, where T is the period between samples of the input sample rate and n=3.

The four multiplexers of network 520 multiplex between the two sets of upsampled delayed inputs from filters H0, H1 and H2 of unit 510 in response to the two most significant bits of the position index input signal (described in connection with Figures 1 and 4). The two most significant bits identify the sample set from unit 510 that spatially encompass the position of the converter output sample currently being determined. Specifically, the two most significant bits identify and select the sample set comprising the two upsampled delayed output samples located either side of the output sample being determined.

The architectures of Figures 1-3 and 5 are not exclusive. Other architectures may be derived in accordance with the principles of the invention to accomplish the same objectives. In addition, a wide variety of advantageous filter functions may be derived using the disclosed structural factoring method for decomposing filter coefficient matrix expressions. Further, the generalized delay line concept may be used to provide cost-effective universal, flexible sample rate conversion from an input sample rate to virtually any output sample rate.

The disclosed sample rate converter and filter functions may be implemented in whole or in part within the programmed instructions of a microprocessor or other controller. Also, the inventive principles disclosed may be used for any filtering or sample rate converter application including, for example, conversion between different video standards, multi-media composite image formation, data processing for digital data storage such as for CDROM or DVD applications and digital demodulation.